The metrics for a code fragment

№	Name	Description / Comment
1–62	Number of keywords: *continue, for, new, switch, assert, synchronized, boolean, do, if, this, break, double, throw, byte, else, case, instanceof, return, transient, catch, int, short, try, char, final, finally, long, strictfp, float, super, while* (**total** and **average among lines of code** for each)	Represents the contents of the code fragment
63–64	Number of characters in the fragment (**total** and **average among lines of code**)	Represents the size of the fragment
65	Number of lines in the fragment	Represents the size of the fragment
66–67	Nesting area of the fragment (**total** and **average among lines of code**)	Represents the complexity of the fragment
68–69	Number of characters in the enclosing method (**total** and **average among lines of code**)	Represents the size of the enclosing method
70	Number of lines in the enclosing method	Represents the size of the enclosing method
71–72	Nesting area of the enclosing method (**total** and **average among lines of code**)	Represents the complexity of the enclosing method
73–74	Number of references to fields from the enclosing class (**total** and **average among lines of code**)	Represents the coupling with the enclosing class through its fields. Important when copying from another class.
75–76	Number of calls of methods from the enclosing class (**total** and **average among lines of code**)	Represents the coupling with the enclosing class through its methods. Important when copying from another class.
77–78	Coupling with the enclosing class (**total** and **average among lines of code**)	The sum of the two metrics above